Probabilistic Models: Spring 2014
Structure Learning with Dynamic Programming Example

We are given the following local scores for some decomposable scoring function and dataset $\mathcal{D}$.

$Score(A, PA_A : \mathcal{D})$	PA_A
44.10	C E
43.55	B E
46.90	E
47.93	D
45.66	B
48.17	$\emptyset$

$Score(B, PA_B : \mathcal{D})$	PA_B
15.11	C E
8.40	C D
30.90	D
25.37	C
31.08	A
24.11	A C
33.59	$\emptyset$

$Score(C, PA_C : \mathcal{D})$	PA_C
8.40	B E
25.23	A E
8.40	B D
28.04	E
30.90	D
40.25	B
48.47	$\emptyset$

$Score(D, PA_D : \mathcal{D})$	PA_D
8.40	B E
8.40	B C
10.90	E
25.37	C
40.25	B
42.70	A
24.11	A C
42.94	$\emptyset$

$Score(E, PA_E : \mathcal{D})$	PA_E
13.81	B C
12.47	D
24.07	C
43.24	A
17.95	A C
44.51	$\emptyset$

1. Calculate the score of the optimal network according to these scores. Assume we want to minimize the network score.

Useful Algorithms

Notation

- $Score(\mathbf{U})$. The score of the optimal subnetwork over variables $\mathbf{U}$
- $BestScore(X, \mathbf{U})$. The score of the best parent set for X which is a subset of $\mathbf{U}$
- $|\mathbf{U}|$. The number of variables in $\mathbf{U}$

```
procedure EXPAND(node U, sorted family scores BestScore)
    for each leaf in V \ U do
        newScore ← Score(U) + BestScore(leaf, U)
        if newScore < Score(U ∪ leaf) then
            Score(U ∪ leaf) ← newScore
        end if
    end for
end procedure

procedure MAIN(variables V, sorted family scores BestScore)
    Score(∅) ← 0
    for layer l = 0 to |V| do
        for each node U such that |U| = l do
            expand(U, BestScore)
        end for
    end for
    return Score(V)
end procedure
```

Note. The score of a subnetwork can also be expressed recursively. The algorithm implicitly calculates this recurrence.

$$Score(\mathbf{U}) = \min_{leaf \in \mathbf{V} \setminus \mathbf{U}} Score(\mathbf{U} \setminus leaf) + BestScore(leaf, \mathbf{U})$$